LECTURE 1

Webpage: [www.cs.ubbcluj.ro/~rlupsa/edu/](http://www.cs.ubbcluj.ro/~rlupsa/edu/)

FINAL GRADE:

60% - the final exam, which takes place during the last week of the semesters

40% - laboratory work

Need at least 4.5 to pass in both the written exam and the lab.

There will be a MS Teams for all the assignments and labs n stuff.

Lab:

* Some lab assignments have bonuses
* You get penalty if u don’t present a lab in time. 2pts / week, max 2 weeks
* At most 2 homeworks can be presented at one lab.
* Practical work 1 => week 5/6
* Practical work 2,3,4,5 on the website

Read the glossary on the website.

Why do we use graphs?

The basic idea is that we discard the irrelevant from a certain problem and we only keep the relevant information. That’s why we use numbers.

We use graphs to get from a point to another. (in general)

We discard the actual road, but we keep in mind the locations we go through. If there is a road between there and we can decide upon that. You can pick the shortest route, or a route that contain certain locations.

Ex.: if u need to install a certain package on your pc, a package manager has to get all the other necessary packages that the one you need depend on, like a binary tree(my op)

Graphs can show not just a ‘road’, but also certain dependencies / requirements. X depends on y depends on z depends on…

G= (X,E)

X := set of vertices, nodes (a finite, non empty set)

(X is non empty and finite for simplicity of theorems)

E := edges

There are 2 types of graphs and 2 types of multigraphs.

1. Directional graphs (x, y) , E is a subset of X \* X

* E is a set; it doesn’t have an order; It doesn’t contain duplicates, each edge appears at most once.
* Loops are allowed.

1. Undirectional graphs {x, y}

* E is a subset of {(x, y)| x, y e X, x!=y}
* Loops aren’t allowed.

The 7 bridges of Konisburg – PROBLEM

* You have a multigraph in this case.

Accessible – a vertex y is accessible from x if there is at least one walk starting at x and ending at y; walks can have length 0 => any vertex is accessible from itself

Adjacency – vertex y is adjacent to vertex x if there is an edge from x t y

Close walk – it’s a walk that starts and ends on the same vertex

Cycle - a closed walk of length >= 1, **no repeating vertexes** except the first one(which is also the last one) and **with no repeating edges**

- in an undirected graph, a cycle has the length at least 3

- there are no cycles in a tree